



SEVENTH FRAMEWORK PROGRAMME Networked Media

Specific Targeted Research Project

SMART

(FP7-287583)

Search engine for Multimedia environment generated content

D2.2 Use Cases Specifications

Due date of deliverable: 30-06-2012

Actual submission date: 03-08-2012

Start date of project: 01-11-2011

Duration: 36 months

Summary of the document

Code:	D2.2 Use Cases Specifications v1.0
Last modification:	26/07/2012
State:	Final
Participant Partner(s):	ATOS, AIT, IBM, Imperial, S3LOG, TELESTO, UG, PRISA, SDR
Author(s):	Antonis Litke (TEL), Karim Moumene Prieto (Prisa), Massimiliano Tarquini (S3Log), John Soldatos (AIT), Maurizio Morgano (S3Log), Dimitris Drakoulis, (TEL), Tomás García Fresno (SDR), Fotini Papavasiliou (TEL), Jose Miguel Garrido (ATOS)
Fragment:	No
Audience:	<input checked="" type="checkbox"/> public <input type="checkbox"/> restricted <input type="checkbox"/> internal
Abstract:	<i>This document illustrates the main use cases that will be designed, implemented and integrated in order to validate, showcase and evaluate the SMART Search Engine. Two main use cases are described and analyzed in the areas of «live news» (media/publishing industry) and «security/surveillance» (security systems industry). Furthermore, the deliverable describes a set of more elementary use cases comprising the above-mentioned applications. At the same time, a use case associated with the proof-of-concept implementation of the SMART architecture is provided.</i>
Keywords:	Search Engine, Multimedia, Live News, Security Surveillance, Use Cases
References:	Deliverable D2.1; DoW



Table of Contents

1	Executive Summary	5
1.1	Scope	5
1.2	Audience	5
1.3	Structure.....	5
2	Introduction.....	6
3	SMART Use cases – overview description	7
3.1	SMART Proof-of-Concept	7
3.1.1	Scenario: What is happening.....	7
3.2	Live News.....	9
3.3	Security	11
4	Use case analysis approach	11
5	Detailed description of SMART Use Cases	13
5.1	SMART Basic Use Cases	13
5.1.1	UC0.1 Log-in.....	13
5.1.2	UC0.2 Register and Deploy a Data Stream.....	14
5.1.3	UC0.3 Manage a Data Feed.....	15
5.1.4	UC0.4 Configure Edge Node	15
5.1.5	UC0.5 Query SMART	16
5.2	Live news	17
5.2.1	Overview	17
5.2.2	UC1.1 Request data: social density.....	18
5.2.3	UC1.2 Request data: social trends	19
5.2.4	UC1.3 Request data: sentiment analysis	20
5.2.5	UC1.4 Request data: Crowds	21
5.2.6	UC1.5 Request data: Events	22
5.2.7	UC1.6 Request data: Color tendency	23
5.2.8	UC1.7 Request data: Local activities agenda.....	24
5.2.9	UC1.8 Read Eskup news and posts	24
5.2.10	UC1.9 Call the Smart Voice mail	25
5.2.11	UC1.10 Post in Eskup.....	27



5.3	Security	28
5.3.1	Overview	28
5.3.2	UC 2.1 Access to Main Panel	29
5.3.3	UC 2.2 Access to Configuration Panel	30
5.3.4	UC 2.3 Get Location sensor list	31
5.3.5	UC 2.4 Check for registered connection	32
5.3.6	UC 2.5 Get sensor working status	33
5.3.7	UC 2.6 Check for events	34
5.3.8	UC 2.7 Pan Zoom tilt Flip Map	35
5.3.9	UC 2.8 Show detail view	36
5.3.10	UC 2.9 Accessing to Administration Panel	37
5.3.11	UC 2.10 Change User Profile	38
5.3.12	UC 2.11 Register/Unregister new SMART Server	39
5.3.13	UC 2.12 Crowd Analysis	40
5.3.14	UC 2.13 Flow Analysis	41
6	Use Case Analysis	43
7	Conclusions	46
8	BIBLIOGRAPHY AND REFERENCES	47



1 **Executive Summary**

1.1 **Scope**

The main objective of the SMART project is to build an open source multimedia search engine, which could provide scalable search capabilities over environment generated content i.e. content captured by the physical world via sensors. A main part of the project will be allocated to the processing of multimedia content derived from visual and acoustic sensors (notably cameras and microphones). The purpose of this processing is to extract pieces of information about the surrounding environment of the sensors based on leading edge audio/visual (A/V) processing components. Such components will be employed in order to allow the SMART systems to perceive the status of the surrounding environment and accordingly to make this context available to the search engine for (later) indexing and retrieval, as required by the SMART applications.

A key prerequisite for the design of the SMART system is the elicitation of requirements from the stakeholders that comprise its value chain. After eliciting the requirements, the SMART consortium has to define the actual use cases that will be implemented within the project so that these can lead to the final definition of the system specification and architecture.

The purpose of the present deliverable is to elaborate on the SMART use cases analysis as far as it concerns how the SMART system will be used from the final users.

1.2 **Audience**

The primary audience of this document consists of the people that will participate in the design of the SMART system. Primarily, the audience concerns members of the consortium who need to undertake the tasks of designing the components and modules of the SMART system. Additionally, the document is of wider interest to stakeholders that are active in multimedia search initiatives, including researchers participating and contributing to FP7 projects under the «'Networked Media and Search Systems» objective.

1.3 **Structure**

This document is structured as following:

- *Section 2* provides the introduction of the deliverable.
- *Section 3* describes the two high level use cases that will be deployed in SMART: Live News and Security.
- *Section 4* gives the analysis approach that is adopted in SMART.
- *Section 5* provides a detailed analysis of the SMART use cases.
- *Section 6* concludes the document.



2 Introduction

This deliverable provides a detailed definition and specification of the validating use cases of the project in the areas of news, security surveillance and more. This deliverable reflects the results of task T2.3 in terms of scenarios and use case specifications.

SMART is developing a multimedia search engine, which will empower searches over multiple repositories of multimedia information that will be populated on the basis of a wider range of physical sensors and social networks. To this end, the project is designing a novel search architecture emphasizing versatile information acquisition from multiple physical and virtual sensors, as well as the fusion of multi-sensory information in accordance to a host of reasoning techniques and rules [Artikis12].

The collection and combination of information from multiple physical and virtual sensors occur within specialized subsystems of the SMART architecture, which are called edge nodes. Another main component of the SMART search architecture is an indexing and retrieval module (based on the Terrier.org search engine), which will be able to index (in nearly real-time) content derived from multiple geographically and administratively dispersed edge nodes. The indexing and retrieval elements of the SMART architecture will facilitate information acquisition associated with social networks and the physical world on the basis of user-defined queries, similar to the way is done by popular search engines (such as Google). Therefore, end-users of the SMART search engine will be able to type/ask queries to the SMART system. At the same time, SMART will provide APIs and related tools empowering the development of search applications (i.e. applications that base their information retrieval capabilities on queries to the SMART search engine). In the scope of the SMART project, we conveniently call these applications SMART applications or SMART use cases.

SMART will be developing, deploying and evaluating two main use cases in the areas of live news and security surveillance domains. These use cases shall be designed and built in a way that will facilitate the understanding, testing and evaluation of the core functionalities and features of the SMART search engine including:

- The ability of the SMART engine to integrate multiple heterogeneous data streams stemming from a variety of physical sensors, virtual sensors, social networks and perceptual processing algorithms.
- The indexing and retrieval of multimedia information from several edge nodes distributed from a geographical and administrative viewpoint.
- The integration of feeds stemming from social networks with data streams stemming from physical sensors deployed in the surrounding environment.
- The event identification and context-aware retrieval functionalities of the SMART search engine.
- The ability of the SMART search engine to leverage information from the LinkedData cloud (c.f. [Heath11]) to facilitate developers and solution providers in the integration of SMART-based solutions.
- The mashup libraries to be developed and provided within SMART in order to facilitate presentation and visualization of the target applications.

The present deliverable is devoted to an in-depth description of the live news and security surveillance description. These use cases will be used to demonstrate the ultimate operation of the fully

fledge SMART system in the scope of the project. They can be however considered as the final target applications for the SMART project. In order to develop and deploy those applications, the consortium intends to focus (first) on the deployment of more basic use cases dealing with atomic functionalities of the SMART system. Furthermore, SMART intends to adopt an iterative and evolutionary approach as part of its software development methodology, which will emphasize the continuous and incremental development and improvement of all the target use cases.

The first step in this evolutionary approach is the development of a proof-of-concept use case, which will ensure that the main principles of the design of the SMART system are robust and feasible. Given the importance of both basic/elementary use cases and of the proof-of-concept outlined above, the present deliverable includes also the description of these use cases.

3 SMART Use cases – overview description

This section provides an overview of the use cases that will be deployed in the SMART project. It starts with the SMART proof of concept setup, which is an indicative setup of the SMART major components in order to design & develop early enough the desired SMART functionality.

3.1 SMART Proof-of-Concept

The purpose of this proof-of-concept setup is to demonstrate all three layers of SMART functionality. Two different edge nodes are generating metadata, the search layer is indexing them and at the application layer mash-ups are provided and queries are answered. An indicative logical structure is presented the figure below. It has to be underlined that the figure below is provided only for consistency reasons in order to allow the reader to have a more clear view of the use cases that will be described in the sequel. The overall SMART architecture will be provided in the deliverable D2.3.

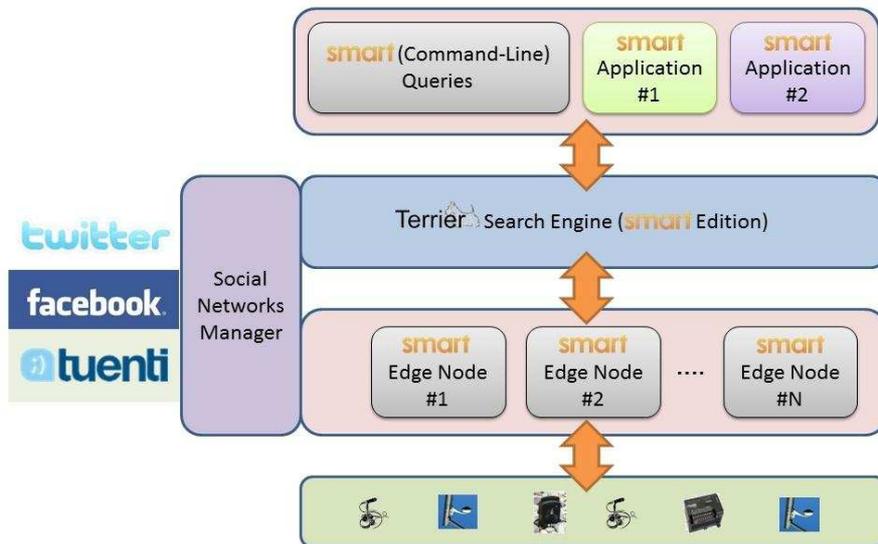


Figure 1 - SMART Layered Architecture

3.1.1 Scenario: What is happening

In a given day, there is a conference at Athens Information Technology, while work is as usual in the various labs.

The conference has attracted some attention in the social media, and there are real-time posts about it. The laboratory also has some minimal tweets associated to it, related to the meetings and demonstrations that are expected to take place that day.

There are two edge nodes, one at the main amphitheatre of Athens Information Technology (@AITathens) and another at the Smart Lab of Athens Information Technology (@AITSmartLab).

@AITathens

At the SMART Demonstration Day conference, identified in Twitter by #SMART_demo, there are four presentations taking place during the afternoon session. They are scheduled as follows:

Time	Speaker	Title
10:00	@jsoldatos	SMART architecture
10:45	@APnevmatikakis	SMART edge node
11:45	@aste80	Querying SMART
12:15	@menelaosbgr	SMART Applications

All these have been tweeted about in the previous days, and there are some real-time tweets by the audience.

The actual times of the four presentations varies a bit, but can be derived by the activity in the conference amphitheatre. From a simple thresholding of a filtered version of the crowd density metric they are:

Scheduled time	Actual timespan
10:00	10:00-10:35
10:45	10:48-11:43
11:45	11:45-12:10
12:15	12:15-12:33

@AITSmartLab

At the AIT SmartLab there are some scheduled events, already announced by tweets:

Scheduled time	Event	Actual timespan
7:10	MeetYourFuture visit group A	7:14-7:26
7:25	MeetYourFuture visit group B	7:28-7:51
8:00	Visit by @PetrosKokk	8:05-8-17
10:45	Conference call	10:52-11:41

There are no real-time tweets about these events. Their actual timespan is again calculated using the crowd analysis metric. There are other unscheduled events (ad hoc group meetings) taking place as well.

Metadata

The systems under test are assumed to provide the following metadata:

- Crowd density metric as continuous visual metadata (i.e. a function of time, one sample per each processing epoch).
- Crowd events: These are visual low-level events resulting from thresholding the crowd density continuous metadata stream. The moment the threshold is exceeded, the crowd event is happening, while the moment the stream receives a smaller value, the crowd event is not happening.

To ensure richness of the metadata, we will also be simulating applause events, i.e. an audio low-level event.

The continuous metadata are mainly for within the edge node consumption, but at least for this proof-of-concept they will be indexed by the search engine.

The low-level events are the inputs of the reasoning engine to provide higher-level events. These low-level events will be passed offline to the reasoning engine in the form of RDF triples. They will also be made available for indexing by the search engine.

Since the two events are actually recorded during two different days at the same location, we need to time-shift the metadata and change the node name of the lab data.

Twitter data

All partners are expected to participate in the tweets announcing the conference event. The hash-tag of the event (#SMART_demo) should be associated to the place it is happening (@AITathens). The tweets should go down to the presentation level of detail, announcing the time schedule.

At the day of the conference, AIT will be generating the real-time tweets.

AIT will also be generating the tweets about the events at AIT Smart Lab.

SMART Mashup and queries

Concerning the mashup, the two edge nodes are intended to be shown on a map, tweets associated to them are streamed and the time intervals of activity are highlighted. The type of activity derived from the reasoning engine can be also included,

Some example queries are:

- What is happening at @AITathens/@AITSmartLab now?
- Has the “Querying SMART” presentation of #SMART_demo started yet? (no/yes/finished)
- Are the MeetYourFuture people finished with @AITSmartLab?

3.2 Live News

The main goal in the Smart Live News use case is to build a tool for citizen (final user) to display geolocalized information, data and statistics about their city. Data sources are going to be both physical world and internet, including social networks and news feeds.

On this use case there are going to be two actors, one is the final user (Citizen) and the other is the Journalist working for the Smart project.

The information is presented in an interactive player. The final user is going to be able to retrieve information by interactions on the player, selecting the type of information wanted to be displayed.

The information that final users can requests interacting with the player is:

- Social density conversation (twitter)
 - Represented on an interactive map.
 - The number of tweets on a concrete location will be displayed.
 - Give an idea of the areas with more social conversation activity depending on the date/time.
- Social conversation trends (#twitter)
 - Represented on an interactive map.
 - Trends represented as a Geolocalized words cloud.
 - Relevant topics are highlighted.
- Social conversation sentiment analysis
 - Represented on an interactive map.
 - Use of colors for representing sentiments.
- Events registered
 - Represented on an interactive map.
 - Events detected from sensors.
 - Inferred cause of the events detected.
- Color tendency
 - Represented on an interactive map.
 - The objective is to make available for the user the color tendency (clothes) of his city depending on location and date/time. In some countries, (for instance, in Spain) it is quite typical that demonstrators wear the same colour.
- Local agenda
 - Represented on a list or a calendar.
 - List of relevant events for the city extracted from the City Hall agenda.
- Eskup news and posts
 - List the posts and news published by users or journalists on Eskup.
 - News categorized, displayed filtered by category.
- Local breaking news:
 - Breaking news are presented as popups when detected.

The second actor that can interact with the system is a journalist working for the Smart Project. The journalist is going to interact with tools to publish news items on Smart.

- Post in Eskup
 - Both Citizen and Journalist are going to be able to post on Eskup.
 - Login is requested to post.
 - Used for contributing with information or comments about events happened on the city.
- Call the smart Voice mail:
 - Only for journalists collaborating with the Project.
 - The Journalist calls the smart voice mail to publish a news Item.
 - Voice recognition is used to authorize the publication.
 - Voice transcription is used to publish the message as a text.

3.3 Security

Security use case target is building a tool capable of getting data from SMART systems and graphically build queries to be submitted. The scope is to query to determine anomalous events that could represent security scenarios.

These scenarios deal with the aggregation and process data looking for sudden variations compared to average data.

- Too many people in a region of interest;
- Too many people in a region of interest mediated by temporal window: number of people is increasing or decreasing too quickly;
- Medium speed of people is superior or inferior to a specific threshold;
- The average speed in a region of interest undergoes a sudden change;
- A people crosses the line in a specified direction (prohibited direction);
- A people crosses the line in a direction that is opposite to the direction used by the most of people;
- The number of people flowing undergoes a sudden change.

All the activities of querying SMART Server will be done in a graphical mode.

In addition to the security scenarios, this use case attempts to demonstrate how SMART different nodes can be dynamically coupled to create a single network for the monitoring of abnormal events.

This use case takes care of two different actors: *User* and *Administrator*. User activities are a subset of Administrator activities. User is typically a security operator which needs to monitor a location during an event. The second one is the system administrator, capable of setting system preference and, particularly, adding a new “SMART Server” node to the environment.

4 Use case analysis approach

In SMART project, we adopt the following approach to analyse use cases:

1) *Identify the Actors*

- Actor: Someone or something that interacts with, or uses, the system to achieve a desired goal. An actor may be a user of the system. In SMART we describe the user by his/her role.

2) *Identify the Goal*

- Goal: The aim, or purpose of goals summarize system functionality in terms of use from a business perspective. Goals can be identified by:
 - From the high-level scenario
 - By brainstorming
 - By asking “What does this Actor want to do?”

3) *Define the Pre-Conditions*

- Pre-Condition: Something that must happen before the Use Case can start; something that must be in place before the Use Case can start. Identify Pre-Conditions by asking:



- "What must be in place for the Use Case to begin?"
- "How do you know you need to do this?"

4) *Define the Post-Conditions*

- Post-Conditions: The result, or successful outcome of the Use Case. Identify the Post-Conditions by asking: "What is the successful result of this process or Use Case?"

5) *Describe the Main Flow*

- Main Flow: Also known as Primary Scenario. A single thread of events where everything goes as defined. A primary scenario starts with pre-conditions and ends with post conditions. The main flow can be described by asking:
 - "What must happen to achieve the goal / outcome?"
 - "What does the actor need to do next?"
 - "What might happen next?"
 - "What do you need to do to get from the trigger to the outcome?"

6) *Describe the Exceptions*

- Exceptions are created to describe the failure situations. In this case all necessary conditions have to be identified where a failure can occur.

5 Detailed description of SMART Use Cases

5.1 SMART Basic Use Cases

In the current section, a first analysis of the basic use cases is presented. All these use cases are common to the proposed SMART Use Cases: Live News and Security. The following figure gives an overview of the SMART Basic use cases.

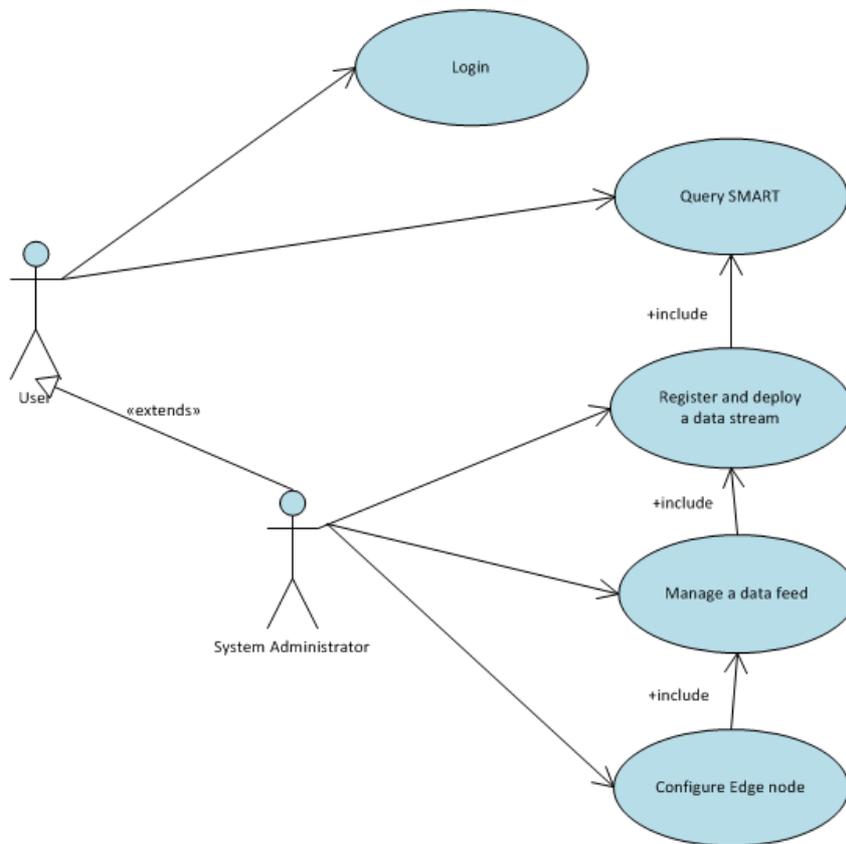


Figure 2: Overview of the SMARTBasic Use Cases (UML Diagram)

5.1.1 UC0.1 Log-in

The purpose of this use case is to log in into the system and have access to the offered functionality.

Name	<i>Log-in</i>
Identifier	<i>UC0.1</i>
Description	<i>SMART user logs into the system.</i>

Goal	<i>Allow the usage of the SMART functionality as offered to its users.</i>
Scope	<i>Within the overall SMART system and its use cases</i>
Preconditions	<i>The user logs in by giving his/her credentials. The user should already have registered into the system.</i>
Post conditions	<i>The user is logged in and can access the SMART services.</i>
Actors	<i>SMART User</i>
Included Use Cases	
Exceptions	<i>The user does not provide correct credentials. In this case the system shows a warning message. After three attempts the system is closed.</i>
Notes/Comments	<i>The initial use case for every subsequent usage of the system.</i>

Table 1: Log in use case

5.1.2 UC0.2 Register and Deploy a Data Stream

The purpose of this use case is add into the SMART system a new data stream source that will be used in order to feed the system with its data.

Name	<i>Register a Data Stream</i>
Identifier	<i>UC0.2</i>
Description	<i>Add into the SMART system a new data stream source that will be used in order to feed the system with its data</i>
Goal	<i>Enhance SMART functionality with new data streams (audio-visual and non audiovisual as well as data from social networks).</i>
Scope	<i>Within the overall SMART.</i>
Preconditions	<i>The system may have other data streams already registered.</i>
Post conditions	<i>The system has a new data stream whose data are feeding now its mechanisms.</i>
Actors	<i>SMART administrator</i>
Included Use Cases	
Exceptions	<i>Warning if the data stream cannot be registered.</i>
Notes/Comments	<i>The initial use case for every subsequent usage of the system.</i>

Table 2: Register and deploy a new data stream use case



5.1.3 UC0.3 Manage a Data Feed

The purpose of this use case is to manage the data feeds of the SMART system as they are coming from the various registered data sources.

Name	<i>Manage a Data Feed</i>
Identifier	<i>UC0.3</i>
Description	<i>Manage the data of a SMART data stream source.</i>
Goal	<i>To allow various management operations on the data feeds such as apply filters, apply reasoning, transform into another format, store, retrieve etc.</i>
Scope	<i>Within the overall SMART functionality.</i>
Preconditions	<i>The system must have a data stream already registered.</i>
Post conditions	<i>Managed and processed data as output.</i>
Actors	<i>SMART administrator</i>
Included Use Cases	<i>UC0.2</i>
Exceptions	
Notes/Comments	

Table 3: Register and deploy a new data stream use case

5.1.4 UC0.4 Configure Edge Node

The purpose of this use case is to allow a proper configuration of the Edge Node in order to make it functional according to the desired use cases.

Name	<i>Configure Edge Node</i>
Identifier	<i>UC0.4</i>
Description	<i>Configure the edge node according to the SMART desired functionalities.</i>
Goal	<i>To allow an easily defined configuration and adaptation mode. Configuration will be performed in high level mode with the appropriate configuration files, that will denote various reasoning parameters, operation mode etc.</i>
Scope	<i>Within the overall SMART functionality.</i>
Preconditions	<i>The system must have a data stream already registered.</i>

Post conditions	<i>Edge node is configured as expected.</i>
Actors	<i>SMART administrator</i>
Included Use Cases	<i>UC0.2, UC0.3</i>
Exceptions	<i>Wrong configuration parameters will result in malfunction of the edge node and therefore warning messages have to be prompted whenever the values are out of range.</i>
Notes/Comments	<i>It is necessary to have an easy configuration mode, with high abstraction so that it gets easy to change parameters and have a very flexible and manageable edge node. Configuration parameters will include also filters data received for the social networks.</i>

Table 4: Configure edge node use case

5.1.5 UC.5 Query SMART

The purpose of this use case is to query the SMART system and retrieve the desired results from the SMART system.

Name	<i>Query SMART</i>
Identifier	<i>UC0.5</i>
Description	<i>Query the SMART search engine in order to retrieve the desired multimedia results.</i>
Goal	<i>The goal of this use case is to allow the user perform search queries to the SMART system and have the ranked results as it is the general case with the search engines.</i>
Scope	<i>Within the overall SMART functionality.</i>
Preconditions	<i>The system must have an already populated data base from which the results will be retrieved.</i>
Post conditions	<i>Search results are presented to the user, or if no results match the query then the results list will be empty.</i>
Actors	<i>SMART user</i>
Included Use Cases	<i>UC0.2, UC0.3, UC0.4</i>
Exceptions	
Notes/Comments	

Table 5: Query SMART use case

In the sequel, the specific use cases for Live News and Security are presented.

5.2 Live news

5.2.1 Overview

This section will analyze the use cases that belong to the overall Live News use case as defined in the SMART project. The objective of this use case is to create a container for local news and general information about the city of Santander. This container could be a website where a Santander's citizen can find information and statistics about what is happening (and happened) around him, e.g. in his/her neighbourhood or his city. The objective is to build it using:

- Automatic event extraction by processing the streams from the deployed sensors.
- Social networks conversation, including other user comments on Eskup.
- City hall events agenda.
- Breaking news from Eskup (from the EIPais Eskup's journalists).
- News coming from other local news RSS feeds.

The following UML figure gives an overview of the specific use case:



Figure 3: Overview of the Live News Use Case (UML Diagram)

5.2.2 UC1.1 Request data: social density

Description: The user requests social density information, the information requested is the geolocalized density data of social conversation (for example twitter) displayed on a Map or timeline view.

Name	Request data: <i>social density</i>
Identifier	<i>UC1.1</i>
Description	<i>Smart final user requests social density information, the information requested is the geolocalized density data of social conversation (for example twitter) displayed on a Map or timeline view.</i>
Goal	<i>Display geolocalized social conversation density on a map view.</i>
Scope	<i>Within the overall Live News use case.</i>
Preconditions	<i>The user selects the social density window on the Smart Player</i>
Post conditions	<i>The information requested is displayed</i>
Actors	<i>SMART final User</i>
Included Use Cases	<i>Switch to timeline view (Display social density data on a timeline view)</i>
Exceptions	
Notes/Comments	<i>The user accesses this functionality through the main smart interactive player.</i>

Table 6: Request data: social density

The purpose of UC1 is to actually give the opportunity to the end-user to find relevant information on social activities that take place in the area of his interest.

The table below gives the flow of events for UC1.1.

Flow of Events
1. Click in the social data density window
2. Display social density data on a Map
3. Drag and drop on map to view another area on the map
4. Update the social density data for the view selected by the user
5.if Click in the timeline button to see the data on a timeline view

5.1. Display social density data on a timeline View
5.2.if Click in Map View
5.2.1.jump 2. SYSTEM Display social density
end if
end if

Table 7: UC1.1 – flow of events

5.2.3 UC1.2 Request data: social trends

Description: The user requests social trends, the information requested is the geolocalized trending topics on social network displayed on a Map or timeline view.

Name	<i>Request data: social trends</i>
Identifier	<i>UC1.2</i>
Description	<i>Smart final user requests social trends data. The information requested is the geolocalized trending topics on social networks displayed on a Map or timeline view.</i>
Goal	<i>Display geolocalized social conversation trends on a map view.</i>
Scope	<i>Within the overall Live News use case.</i>
Preconditions	<i>The user selects the social trends window on the Smart Player</i>
Post conditions	<i>The information requested is displayed</i>
Actors	<i>SMART final User</i>
Included Use Cases	<i>Switch to timeline view (Display social density data on a timeline view)</i>
Exceptions	
Notes/Comments	<i>The user accesses this functionality through the main smart interactive player.</i>

Table 8: Request data: social trends

The purpose of UC1.2 is to actually give the opportunity to the end-user to find relevant information on social trends through specific social networks and the discussions covered there..

The table below gives the flow of events for UC1.2.

Flow of Events Error! Bookmark not defined.
1. Click in the social trends window
2. Display social trends on a Map View
3. Drag and drop on map to view another area on the map
4. Update the social trends data for the view selected by the user
5.if Click in the timeline bouton to see the data on a timeline view
5.1. Display social trends data on a timeline View
5.2.if Click in Map View
5.2.1.jump 2. <u>SYSTEM Display social trend...</u>
end if
end if

Table 9: UC1.2 – flow of events

5.2.4 UC1.3 Request data: sentiment analysis

Description: The user request social sentiment analysis (see for example [Agarwal11]), the information requested is the geolocalized sentiments inferred from social conversation.

Name	<i>Request data: sentiment analysis</i>
Identifier	<i>UC1.3</i>
Description	<i>Smart final user requests social sentiment analysis data. The information requested is the geolocalized sentiments inferred from social conversation.</i>
Goal	<i>Display geolocalized social conversation sentiment analysis results of the social conversation around the city.</i>
Scope	<i>Within the overall Live News use case.</i>
Preconditions	<i>The user selects the sentiment analysis window on the Smart Player</i>
Post conditions	<i>The information requested is displayed</i>
Actors	<i>SMART final User</i>
Included Use Cases	<i>Switch to timeline view (Display sentiment analysis results on a timeline view)</i>
Exceptions	
Notes/Comments	<i>The user accesses this functionality trough the main smart interactive player.</i>

Table 10: Request data: sentiment analysis

The purpose of this use case is to give the SMART end user the ability to identify the sentiment and opinion (wherever possible) in the written messages of the social media that his is searching for.

The table below gives the flow of events for UC1.3.

Flow of Events
1. Click in social sentiment window
2. Display social sentiment on a Map View
3. Drag and drop on the map to view another area on the map
4. Update the social sentiment analysis data for the view selected by the user
5.if Click in the timeline button to see the data on a timeline view
5.1. Display sentiment analysis data on a timeline View
5.2.if Click in Map View
5.2.1.Jump 2. SYSTEM Display social senti...
end if
end if

Table 11: UC1.3 – flow of events

5.2.5 UC1.4 Request data: Crowds

Name	<i>Request data: Crowds</i>
Identifier	<i>UC1.4</i>
Description	<i>Smart final user request the information about the crowds location on the city. The data of the geolocalized crowds registered by the smart system displayed on a map.</i>
Goal	<i>Display Crowds detected in the city.</i>
Scope	<i>Within the overall Live News use case.</i>
Preconditions	<i>The user selects the Crowds window on the Smart Player</i>
Post conditions	<i>The information requested is displayed</i>
Actors	<i>SMART final User</i>
Included Use Cases	<i>Switch to timeline view (Display Crowds detected on a timeline view)</i>
Exceptions	
Notes/Comments	<i>The user accesses this functionality trough the main smart interactive player.</i>

Table 12: Request data: crowds

The purpose and role of this use case in the SMART demonstration is to test Video processing algorithm for crowd analysis.

5.2.6 UC1.5 Request data: Events

Name	<i>Request data: events</i>
Identifier	<i>UC1.5</i>
Description	<i>Smart final user request information about events detected through sensors, social networks ([Becker09]), these events could be accidents, shooting etc.</i>
Goal	<i>Display events registered on the city</i>
Scope	<i>Within the overall Live News use case.</i>
Preconditions	<i>The user selects the events window on the Smart Player</i>
Post conditions	<i>The information requested is displayed</i>
Actors	<i>SMART final User</i>
Included Use Cases	<i>Switch to timeline view (Display Events detected on a timeline view)</i>
Exceptions	
Notes/Comments	<i>The user accesses this functionality through the main smart interactive player.</i>

Table 13: Request data: events

The purpose and role of this use case in the SMART demonstration is to test Video and audio processing algorithm for detecting events through audio and video streams.

The table below gives the flow of events for UC1.5.

Flow of Events
1. Click in Events window
2. Display Events on a Map View
3. Drag and drop on map to view another area on the map
4. Update the Events for the view selected by the user
5.if Click in the timeline bouton to see the data on a timeline view
5.1. Display the eventes on a timeline View
5.2.if Click in Map View
5.2.1.jump 2. SYSTEM Display Events on a ...
end if
end if

Table 14: UC1.5 – flow of events

5.2.7 UC1.6 Request data: Color tendency

Description: The user requests the color tendency: the predominant colors on the city detected from the cameras installed and displayed on a map.

Name	<i>Request data: Color tendency</i>
Identifier	<i>UC1.6</i>
Description	<i>SMART end user requests information about the color tendency on his city, Predominant colors (clothes) detected on the city from the cameras installed displayed on a map.</i>
Goal	<i>Display clothes color tendency on the city Test color detection of the video analysis algorithms.</i>
Scope	<i>Within the overall Live News use case.</i>
Preconditions	<i>The user selects the Color tendency window on the SMART Player</i>
Post conditions	<i>The requested information is presented to the user.</i>
Actors	<i>SMART final User</i>
Included Use Cases	<i>Switch to timeline view</i>
Exceptions	
Notes/Comments	<i>The user accesses this functionality through the main smart interactive player.</i>

Table 15: Request data: Color tendency

The purpose and role of this use case in the SMART demonstration is to test the Video processing algorithm for *color detection*.

The table below gives the flow of events for UC1.6.

Flow of Events
1. Click in Color tendency window
2. Display the actual Color tendency on a Map View
3. Drag and drop on map to view another area on the map
4. Update the color tendency data for the view selected by the user
5. Interact with the timeline to see the color tendency on another time, day week month
6. update the map taking into consideration the time/date selected by the user

Table 16: UC1.6 – flow of events

5.2.8 UC1.7 Request data: Local activities agenda

Description: The user requests the local activities agenda, the information requested is the city hall agenda of events for the city, that information will be displayed for the current week, the user can request to display data for a date selected.

Name	<i>Request data: Local activities agenda</i>
Identifier	<i>UC1.7</i>
Description	<i>Smart final user request the information about local agenda activities planned for the city.</i>
Goal	<i>Display city hall activities agenda</i>
Scope	<i>Within the overall Live News use case.</i>
Preconditions	<i>The user selects the local Agenda window on the Smart Player</i>
Post conditions	<i>The information requested is displayed</i>
Actors	<i>SMART final User</i>
Included Use Cases	
Exceptions	
Notes/Comments	<i>The user accesses this functionality through the main smart interactive player.</i>

Table 17: Request data: Local activities agenda

The purpose and role of this use case is to inform the end user on events and activities that are of his interest.

The table below gives the flow of events for UC1.7.

Flow of Events Error! Bookmark not defined.
1. Click in Local Agenda window
2. Display Agenda data for the current Week
3.if Select another date on the calendar.
3.1. Update the agenda data for the week selected by the user
end if

Table 18: UC1.7 – flow of events

5.2.9 UC1.8 Read Eskup news and posts

Description: The user requests to read the news and posts published on Eskup. The user can select the category.



Name	<i>Read Eskup news and posts</i>
Identifier	<i>UC1.8</i>
Description	<i>Smart final user requests to read the news and posts published on Eskup. The user selects posts filtered by category.</i>
Goal	<i>Display news published and user posts related to the city on Eskup</i>
Scope	<i>Within the overall Live News use case.</i>
Preconditions	<i>The user selects the Eskup window on the Smart Player</i>
Post conditions	<i>The information requested is displayed</i>
Actors	<i>SMART final User</i>
Included Use Cases	<i>Select an Eskup Category</i>
Exceptions	
Notes/Comments	<i>The user accesses this functionality through the main smart interactive player.</i>

Table 19: Read Eskup news and posts

The purpose and role of this use case is to inform the end user on news and posts on the Eskup social network.

The table below gives the flow of events for UC1.8.

Flow of Events
1. The user request to open the Eskup window
2. Display the category list
3. the user selects a category
4. Display the category posts list preview
5.if the user clicks on a post
5.1. display the complete post
6.else if the user clicks on back
6.1.jump 4. SYSTEM Display the category...
end if

Table 20: UC1.8 – flow of events

5.2.10 UC1.9 Call the Smart Voice mail

Description: The journalist calls the Smart Voice mail to publish a news Item. The system checks

his identity using voice recognition and records the news item to publish it as text using speech to text transcription.

Name	<i>Call the Smart Voice mail</i>
Identifier	<i>UC1.9</i>
Description	<i>The journalist calls the Smart Voice mail to publish a news Item. The system checks his identity, and publish automatically the news item on Eskup</i>
Goal	<i>Give an quick and automatic way for journalist to publish on Eskup.</i>
Scope	<i>Within the overall Live News use case.</i>
Preconditions	<i>The Actor calls the Smart voice mail</i>
Post conditions	<i>The message is published on Eskup</i>
Actors	<i>SMART Journalist</i>
Included Use Cases	<i>N/A</i>
Exceptions	
Notes/Comments	<i>The Journalist accesses this functionality calling a voice mail.</i>

Table 21: Call the Smart Voice mail

The purpose and role of this use case in the SMART demonstration is to *allow testing*:

- *Voice recognition: for journalist identification*
- *Voice transcription: for converting the audio recored to text.*

The table below gives the flow of events for UC1.9.

Flow of Events
1. The Journalist call the Smart Eskup voice mail
2. the voice mail request the user to say a sentence to identify Smart authorized journalists
3. The journalist speaks to be identified
4.if The identification process (voice recognition) is successful
4.1. Talk after the "beep" to record the news Item
5.else
5.1. Identification failed, you are not authorized to publish as Smart Journalist, please try again the identification process
5.2.jump 2. SYSTEM the voice mail request...
end if

Table 22: UC1.9 – flow of events

5.2.11 UC1.10 Post in Eskup

Description: The user or the Journalist writes a post in Eskup, after identification and category selection.

Name	<i>Post in Eskup</i>
Identifier	<i>UC1.10</i>
Description	<i>Smart final user or Journalist writes a post in Eskup, after identification and category selection.</i>
Goal	<i>Give the user tools to publish news Items and posts about the city in Eskup</i>
Scope	<i>Within the overall Live News use case.</i>
Preconditions	<i>The user has opened the Eskup Window</i>
Post conditions	<i>The post published on Eskup</i>
Actors	<i>SMART final User & SMART Journalist</i>
Included Use Cases	<i>Eskup User login</i>
Notes/Comments	<i>The user accesses this functionality through the main smart interactive player.</i>

Table 23: Post in Eskup

The purpose and role of this use case in SMART is to allow for publication of posts in the Eskup from both categories of users (end –users and journalists).

The table below gives the flow of events for UC1.10.

Flow of Events
1. Click on the button post in Eskup
2.if Not identified?
2.1. Show the identification process requesting User & Password o the possibility to create a new user
2.2.if user introduces a user and password and cick Login
2.2.1.if Login is correct
2.2.1.1.
2.2.1.2.jump 3. SYSTEM Display the categories
2.2.2.else
2.2.2.1.jump 2.1. SYSTEM Show the identification...
end if
2.3.else if User clicks on create your user

2.3.1. Display register formular : Name; nickname; e-mail; age; ...
2.3.2. User introduces the data and click continue
2.3.3.if data are correct
2.3.3.1. Your user has been created
2.3.3.2.jump 3. Display the categories available on Eskup
2.3.4.else
2.3.4.1. Some data are missing or not correct
2.3.4.2.jump 2.3.1. SYSTEM Display registry
end if
end if
end if
3. Display the categories available on Eskup
4. The User selects a category
5. Display the blank space to write the post
6. User writes the post and clicks publish
7. The post has been published correctly

Table 24: UC1.10 – flow of events

5.3 Security

5.3.1 Overview

An important application for the SMART project is the automatic analysis and detection of abnormal events in public places or during public events. One particular class of public security issues are those involving a large number of people gathering together (crowding), such as in public assemblies, sport competitions, demonstrations (e.g., strikes, protests), etc. Because of the high level of degeneration risk, the security of public events involving large crowd has always been of high concern to relevant authorities.

Challenge for this experimentation is automatic detection of abnormal crowd behaviours during public events. We have divided crowd behaviour analysis into three tasks:

- *Task 1: motion information extraction;*
- *Task 2: noise information extraction;*
- *Task 3: abnormal behaviour modelling.*

We are looking for crowd behaviour such as:

- *flow divergence and convergence in the region of interest;*
- *emergence of a new crowd flow from an existing crowd flow having different characteristics;*
- *sudden changes in the crowd motion;*
- *people moving in counter direction;*
- *erratic motion beside the main crowd flow (as in the case of fighting);*
- *sudden change in the noise of the crowd.*

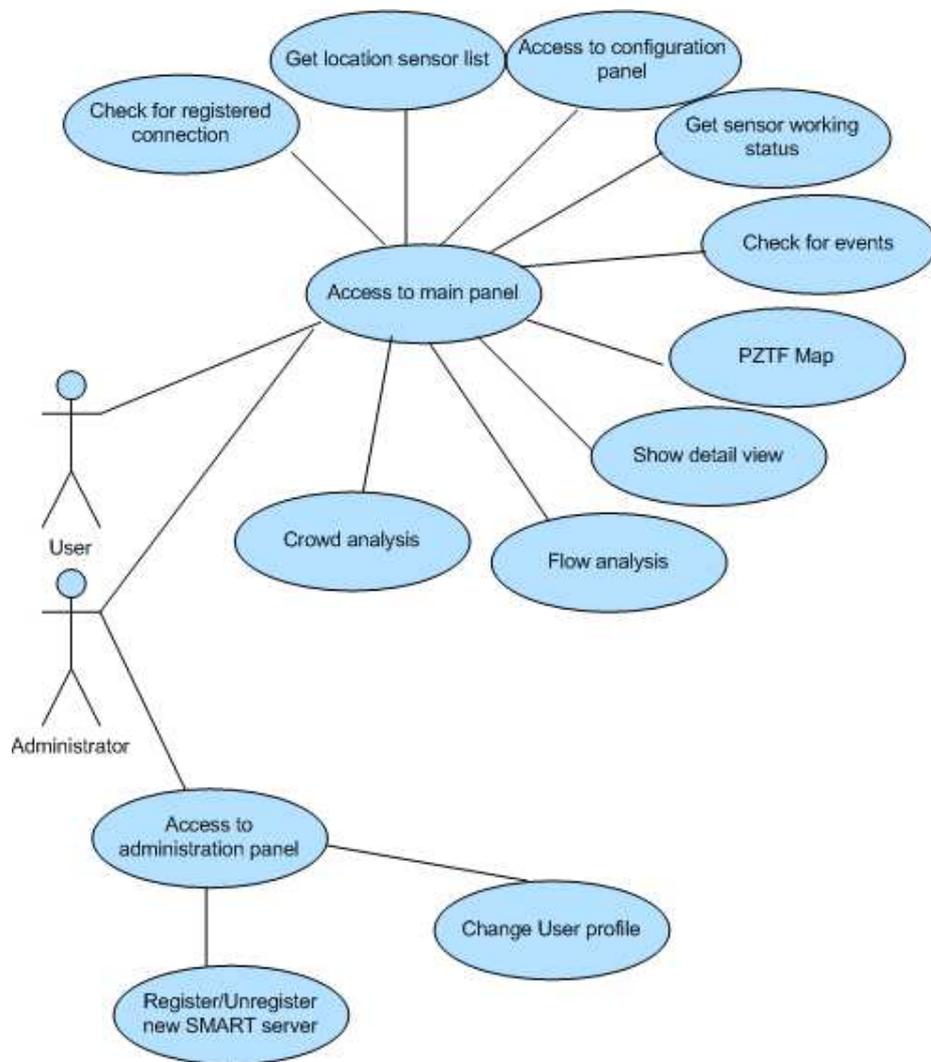


Figure 4: Overview of the Security Use Case (UML Diagram)

5.3.2 UC 2.1 Access to Main Panel

The purpose and role of this use case in SMART is to allow access to main panel of the security application provided by S3Log.

Name	<i>Access to main Panel</i>
Identifier	<i>UC2.1</i>
Description	<i>A smart user needs to make a search using SMART. He logs in to the application and access to the main application panel to start working on.</i>

Goal	<i>Logging into the application</i>
Scope	<i>Within the overall Security use case.</i>
Preconditions	<i>User has the credentials to Login</i>
Post conditions	<i>User has successfully logged in.</i>
Actors	<i>SMART User</i>
Included Use Cases	<i>N/A</i>
Exceptions	<i>Warning in case the credentials are not correct.</i>
Notes/Comments	<i>This UC is for the specific access to the security application main panel.</i>

Table 25: Access to main Panel

The table below gives the flow of events for UC2.1.

Flow of Events
1 start the system and click to open the main window
2 show the login panel
3 logging into the application with username e password
4 show the main panel

Table 26: UC2.1 – flow of events

5.3.3 UC 2.2 Access to Configuration Panel

The purpose and role of this use case in SMART is to allow access to the configuration panel of the SMART system in order to change the settings of the application.

Name	<i>Access to configuration panel</i>
Identifier	<i>UC2.2</i>
Description	<i>A smart user needs to change application working parameters. To do this access to the application configuration panel.</i>
Goal	<i>Changing application working parameters: Interface parameters</i>
Scope	<i>Within the overall Security use case.</i>
Preconditions	<i>User is logged in User has rights to change the settings of the application</i>

Post conditions	<i>Ability to have full access to the configuration panel</i>
Actors	<i>SMART User</i>
Included Use Cases	<i>UC2.1</i>
Exceptions	<i>Warning in case the there are no rights to change the settings.</i>
Notes/Comments	

Table 27: Access to configuration panel.

The table below gives the flow of events for UC2.2.

Flow of Events
1 click to open the configuration panel
2 show the configuration panel

Table 28: UC2.2 – flow of events

5.3.4 UC 2.3 Get Location sensor list

The purpose and role of this use case is to retrieve the list of the various sensors and their location.

Name	<i>Get Location Sensor List</i>
Identifier	<i>UC2.3</i>
Description	<i>In this use case the user drill down into a registered location to get the list of all the sensors connected to an edge server.</i>
Goal	<i>Accessing to the geographical network of sensors and edge server.</i>
Scope	<i>Within the overall Security use case.</i>
Preconditions	<i>User is logged in At least one SMART server is connected</i>
Post conditions	<i>A location list is presented to the user.</i>
Actors	<i>SMART User</i>

Included Use Cases	<i>Is related to "Get Sensor working status" use case</i>
Exceptions	
Notes/Comments	

Table 29: Get location sensor list

The table below gives the flow of events for UC2.3.

Flow of Events Error! Bookmark not defined.
1 click to open the map panel
2 shows the map panel, check for registered connection and update map

Table 30: UC2.3 – flow of events

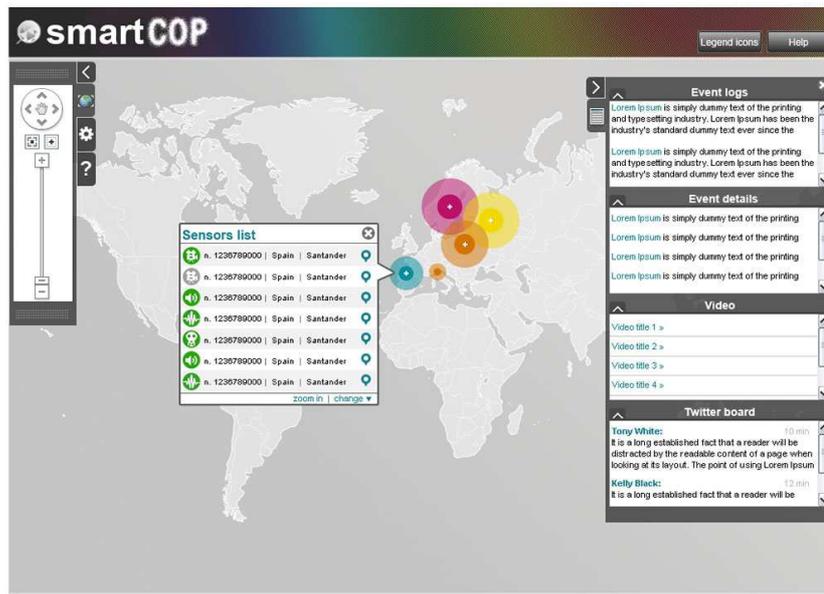


Figure 5: Sensor List

5.3.5 UC 2.4 Check for registered connection

The purpose and role of this use case is to retrieve the list of registered edge servers and sensors and their location.

Name	<i>Check for registered connection</i>
Identifier	<i>UC2.4</i>
Description	<i>In this use case, the SMART user is looking for all the sensors and edge server registered to the SMART System. Each edge server or sensor is displayed on a geographical location basis or listed in an appropriate window.</i>
Goal	<i>Accessing to the geographical network of sensors and edge server</i>
Scope	<i>Within the overall Security use case.</i>
Preconditions	<i>User is logged in At least a SMART server is connected</i>
Post conditions	<i>A location list is presented to the user.</i>
Actors	<i>SMART User</i>
Included Use Cases	<i>UC2.3 Get Location Sensor List</i>
Exceptions	
Notes/Comments	

Table 31: Check for registered connection

The table below gives the flow of events for UC2.4.

Flow of Events
1 click to open the map panel
2 shows the map panel, check for registered connection and update map

Table 32: UC2.4 – flow of events

5.3.6 UC 2.5 Get sensor working status

The purpose and role of this use case is to identify the working status of the sensors.

Name	<i>Get Sensor working status</i>
Identifier	<i>UC2.5</i>
Description	<i>In this use case the user polls a sensor or edge server to ask for operating status and errors signals.</i>

Goal	<i>Getting all the information about the working status of a sensor connected to an edge server</i>
Scope	<i>Within the overall Security use case.</i>
Preconditions	<i>User is logged in At least a SMART server is connected</i>
Post conditions	<i>Working status information is presented to the user.</i>
Actors	<i>SMART User</i>
Included Use Cases	<i>UC2.3, UC2.4</i>
Notes/Comments	

Table 33 Get sensor working status

The table below gives the flow of events for UC2.5.

Flow of Events
1 click to open the map panel
2 shows the map panel, check for registered connection and update map

Table 34: UC2.5 – flow of events

5.3.7 UC 2.6 Check for events

The purpose and role of this use case is to retrieve the list of the various events that have been logged from the sensors on the edge servers.

Name	<i>Checks for events</i>
Identifier	<i>UC2.6</i>
Description	<i>In this use case the user access to the logs of the events coming from the SMART network. Events could be of different types: errors, status, data from server, etc. etc.</i>
Goal	<i>Monitoring events coming from a SMART network</i>
Scope	<i>Within the overall Security use case.</i>
Preconditions	<i>User is logged in At least a SMART server is connected</i>
Post conditions	<i>A list of relevant events is presented.</i>
Actors	<i>SMART User</i>

Included Use Cases	<i>UC 2.4</i>
Exceptions	
Notes/Comments	

Table 35: Check for events

The table below gives the flow of events for UC2.6.

Flow of Events
1 click to open the map panel
2 shows the map panel, check for registered connection and update map
3 click to open the Event log panel
4 show the event log panel

Table 36: UC2.6 – flow of events

5.3.8 UC 2.7 Pan Zoom tilt Flip Map

The purpose and role of this use case is to remotely control basic operations of the maps view.

Name	<i>Pan Zoom tilt Flip Map</i>
Identifier	<i>UC2.7</i>
Description	<i>In this use case the SMART user wants to change the view of the maps and elements on it. To do this the user Pan, Zoom, tilt and Flip maps with tools provided by the application interface.</i>
Goal	<i>Changing the data view</i>
Scope	<i>Within the overall Security use case.</i>
Preconditions	<i>User is logged in</i>
Post conditions	<i>A changed view of the map is presented.</i>
Actors	<i>SMART User</i>
Included Use Cases	
Exceptions	
Notes/Comments	<i>In the context of the SMART project this use case may be prohibited for Privacy Reasons.</i>

Table 37: Pan Zoom tilt Flip Map

The table below gives the flow of events for UC2.7.

Flow of Events
1 click to open the map panel
2 shows the map panel, check for registered connection and update map
3 use command to pan or zoom map

Table 38: UC2.7 – flow of events

5.3.9 UC 2.8 Show detail view

The purpose and role of this use case is to show to the end users various details views of the edge server such as geographical location.

Name	<i>Show detailed view</i>
Identifier	<i>UC2.8</i>
Description	<i>This use case describes how the user can drill down into an edge server to get detailed graphical information about and edge server and the geographical location sensors are working on. This view gives an interactive and auto updated representation of a location and can be used as real time monitor.</i>
Goal	<i>Changing the data view</i>
Scope	<i>User needs to monitor an area based upon graphical representation</i>
Preconditions	<i>User is logged in SMART server is connected</i>
Post conditions	<i>Data view has changed.</i>
Actors	<i>SMART User</i>
Included Use Cases	
Exceptions	
Notes/Comments	

Table 39: Show detail view

The table below gives the flow of events for UC2.8.

Flow of Events
1 click to open the map panel
2 shows the map panel, check for registered connection and update map
3 click to open detail view

Table 40: UC2.8 – flow of events

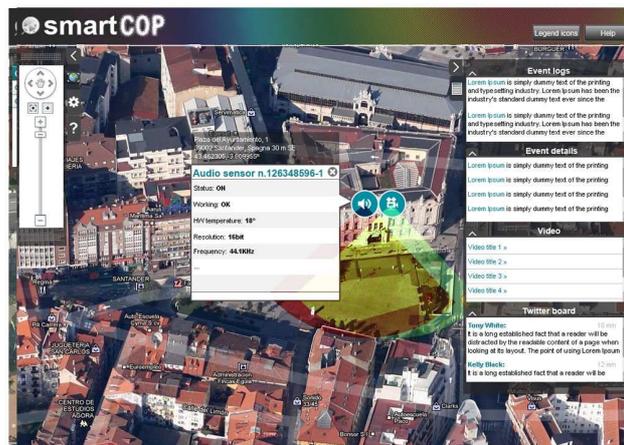


Figure 6: Detail View

5.3.10 UC 2.9 Accessing to Administration Panel

The purpose of this use case is to refer to the capabilities of the administration to access the Administration Panel and change settings regarding the security application.

Name	<i>Accessing to Administration Panel</i>
Identifier	<i>UC2.9</i>
Description	<i>In this use case the administrator access to the administration panel to change working parameters, user profiles, etc. etc.</i>
Goal	<i>New user to be registered User profile to be changed</i>
Scope	<i>Within the overall Security use case.</i>
Preconditions	<i>User is logged in User has administration rights</i>
Post conditions	<i>Full access to the administration panel.</i>
Actors	<i>SMART Administrator</i>

Included Use Cases	<i>UC 2.1</i>
Notes/Comments	

Table 41: Accessing to Administration Panel

The table below gives the flow of events for UC2.9.

Flow of Events
1 click to open the administration panel
2 shows the administration panel

Table 42: UC2.9 – flow of events

5.3.11 UC 2.10 Change User Profile

The purpose and role of this use case allow the SMART end user to change his profile information and access parameters of the service.

Name	<i>Change user Profile</i>
Identifier	<i>UC2.10</i>
Description	<i>A smart user needs to change user profile information and access parameters.</i>
Goal	<i>Changing the user profile data</i>
Scope	<i>Profile data changed Unsecured password Password expired</i>
Preconditions	<i>User is logged in</i>
Post conditions	<i>User profile data have been changed.</i>
Actors	<i>SMART User</i>
Included Use Cases	<i>UC2.9</i>
Exceptions	
Notes/Comments	

Table 43: Change User Profile

The table below gives the flow of events for UC2.10.

Flow of Events
1 click to open the administration panel
2 shows the administration panel
3 click to open the user profile panel

Table 44: UC2.10 – flow of events

5.3.12 UC 2.11 Register/Unregister new SMART Server

The purpose and role of this use case allow for a new registration of a SMART edge server that will be included in the overall system.

Name	<i>Register/Unregister new SMART Server</i>
Identifier	<i>UC2.11</i>
Description	<i>SMART architecture is modular and scalable. The SMARTCop application supports this characteristic by giving the possibility to dynamically register or unregister an edge server. This use case approach to the problem of registering a new server to be used or unregistering an unused server.</i>
Goal	<i>New server available/unavailable</i>
Scope	<i>Within the overall Security use case.</i>
Preconditions	<i>User is logged in User has administration rights</i>
Post conditions	<i>New server has been registered/unregistered.</i>
Actors	<i>SMART User</i>
Included Use Cases	<i>UC 2.9</i>
Exceptions	
Notes/Comments	

Table 45: Register/Unregister new SMART Server

The table below gives the flow of events for UC2.11.

Flow of Events
1 click to open the administration panel
2 shows the administration panel

3 click to open SMART configuration server panel
4 show SMART Server list
5 if add: click to add new SMART Server
5.1 Shows a panel to add a new SMART Server
5.2 Insert new server parameter and click to save
5.3 Check new parameter and save new SMART Server, update server list
End if
5 if delete: click to delete SMART Server
5.1 Remove selected server from server list
End if

Table 46: UC2.11 – flow of events

5.3.13 UC 2.12 Crowd Analysis

The purpose and role of this use case is to perform an analysis on the existence of a crowd in a specific location.

Name	<i>Crowd Analysis</i>
Identifier	<i>UC2.12</i>
Description	<i>The user can start a new crowd analysis.</i>
Goal	<i>User set up query element for crowd analysis</i>
Scope	<i>Crowd analysis within the overall Security use case.</i>
Preconditions	<i>User is logged in</i>
Post conditions	<i>Information of crowd analysis is presented to the user.</i>
Actors	<i>SMART User</i>
Included Use Cases	<i>UC2.1</i>
Exceptions	
Notes/Comments	

Table 47: Crowd Analysis

The table below gives the flow of events for UC2.12.

Flow of Events
1. Open the crowd panel administration
2. Set up the query parameter: <ul style="list-style-type: none"> Area to monitor
3. Save query parameter and start collector data
4. Stop collector data
5 start with data analysis and shows result data

Table 48: UC2.12 – flow of events



Figure 7: Crowd Analysis

5.3.14 UC 2.13 Flow Analysis

The purpose and role of this use case is to perform an analysis on the flow of a crowd in a specific location.

Name	<i>Flow Analysis</i>
Identifier	<i>UC2.13</i>
Description	<i>The user can start a new flow analysis.</i>
Goal	<i>User set up query element for flow analysis</i>

	<i>Flow analysis finish</i>
Scope	<i>Flow analysis within the overall Security use case.</i>
Preconditions	<i>User is logged in</i>
Post conditions	<i>Flow analysis is presented to the user.</i>
Actors	<i>SMART User</i>
Included Use Cases	<i>UC2.1</i>
Exceptions	
Notes/Comments	

Table 49: Flow Analysis

The table below gives the flow of events for UC2.13.

Flow of Events
1. Open the flow panel administration
2. Set up the query parameter
3. Save query parameter and start collector data
4. Stop collector data
5. start with data analysis and shows result data

Table 50: UC2.13 – flow of events



Figure 8: Flow Analysis

6 Use Case Analysis

The selection of the above use cases is oriented towards the definition (and later validation) of the main functionalities and features of the SMART engine. Note that the functionalities and features of the SMART search systems are also driven by the requirements specified in the scope of deliverable D2.1. The use cases presented in this document allow for further disambiguation and refinement of the SMART functionalities. At the same time they facilitate the understanding of the scope and operations of the SMART applications i.e. applications that are built over the SMART search engine based on proper combinations of queries to the SMART systems. The main functionalities of the SMART system that have been highlighted as part of the use cases specification concern the following areas:

- Sensors and Perceptual Components:** The use case descriptions indicate the need for deploying sensors (such as cameras) and perceptual components (e.g., video signal processing components in order to support them). For example, the live news use case indicates the need for a color analysis perceptual component, while the security use case indicates the need for a crowd analysis component. The underlying technologies for these components are being developed in the scope of WP3. Note that the definition of the use cases in this respect has also taken into account the competencies of the consortium in terms of video and audio signal processing. While SMART is designed as extensible in terms of supported sensors and perceptual components (based on the requirements listed in D2.1), the SMART use cases demand the development, deployment and use of specific perceptive technologies. The implementation of these technologies has already commenced as part of the SMART workplan, given the time and effort needed to produce robust A/V processing system. **Error! Reference source not found.** illustrates the sensors and perceptual components that will be developed in support of the above-listed use cases.

Sensors and Perceptual Components Developed and Deployed in SMART	Related Use Cases
Visual Signal Processing for Crowd Analysis	UC2.6, UC2.12, UC2.13

Audio Analysis for Crowd Events Identification/Classification	UC2.6, UC2.12, UC2.13.
Visual Signal Processing for Colour Analysis	UC1.6, UC2.12

Table 51: Sensors and Perceptual Components to be developed and deployed in SMART in support of the Use Cases

- Social Networks:** In addition to sensors and media processing components, the use cases impose a need for SMART applications to become integrated with social networks. As is illustrated in the applications, social networks integration has a dual flavor: (a) on the one hand social networks like Twitter and Eskup can be used as a virtual sensor to the SMART system (i.e. so-called «social sensor») in order to derive and analyze context associated with the physical environment, (b) on the other hand, social networks can be seen as part of the end-users applications (e.g., Eskup is an end-user application where messages can be posted). **Error! Reference source not found.** illustrates the type of social networks integration indicated by the SMART Use cases. The contents of this table should be taken in the design and implementation of the SMART Social Network Manager module (SNM module) of the SMART architecture i.e. a module enabling interfacing and management of interfaces to multiple social networks.
- Mashups:** SMART plans the implementation of a mashup library over its search engine, as a means to facilitating the implementation of the presentation tier of SMART applications. The SMART use cases have indicated a number of commonly used mashups, which will be prioritized in the scope of the mashup library implementation. These mashups are listed in Table 53.

Social Networks to be integrated with SMART	Related Use Cases
Eskup (as end-user application)	UC1.8-1.10
Eskup (as a social sensor)	UC1.1, UC1.2, UC1.3
Twitter (as a social sensor)	SMART Proof-of-Concept, UC1.1, UC1.3 (Request data: sentiment analysis)
Facebook	UC1.1, UC1.3 (Request data: sentiment analysis)

Table 52: Social networks to be integrated with SMART in the scope of the SMART Use Cases

- Third Party Software Packages:** The SMART Use Cases indicate also the need to integrate the search engine with third-party software packages such as the COP interface (by S3LOG). (For instance UC2.12 and UC2.13).
- Databases and Linked Data:** The use cases have also underlines the need to use a number of database (as third party datasets). These datasets include agendas/databases of local events (available by partners PRISA and SDR), as well as Linked Data Sets (notably Linked Data associated with restaurant reviews as part of the SMART proof-of-concept use case).

Mashup and GUI Elements to be used in SMART	Related Use Cases
Web2.0 Map / COP Interface	UC2.12 (Crowd Analysis),



	UC2.13 (Get Location sensor list), UC1.5.
Geolocation Map	UC2.12, UC2.13

Table 53: Mashups need for visualization and user interaction in the scope of the SMART use cases



7 Conclusions

Use case analysis is very important for the definition of the final architecture of the SMART system and its specification. Moreover, use cases play a significant role in the validation of the end system with respect to the desired functionality and the usage scenarios.

In this document we have presented the SMART use case analysis from the overall approach to specific use cases that are related to the SMART usage scenarios. A general proof-of-concept scenario is presented that has been designed by the consortium partners in order to early enough allow the design of the desired functionalities of the SMART project. The focus is given on the two major Use Case suites that will be deployed in the project, namely, the *Live News application* and the *Security application*. The document has concluded with a short analysis of the needed sensors, data streams, mashups and third party software as deduced through the use case analysis.

The current deliverable will be one of the main reference documents for the design and specification of the SMART system. In particular, the deliverable D2.3 Multimedia Search Framework Open Architecture and Technical Specifications will have the current deliverable as input.



8 BIBLIOGRAPHY AND REFERENCES

[Agarwal11] Apoorv Agarwal, Boyi Xie, Ilia Vovsha, Owen Rambowod, Rebecca Passonneau, «Sentiment analysis of Twitter data», in the Proceedings of the Workshop on Languages in Social Media (LSM '11), pp. 30-38

[Artikis12] Artikis A., Sergot M. and Paliouras G. Run-Time Composite Event Recognition, International Conference on Distributed Event-Based Systems (DEBS), ACM, 2012.

[Becker09] Becker H., Naaman M., Gravano L., “Event Identification in Social Media”, In WebDB, 2009.

[Heath11] Tom Heath and Christian Bizer (2011) Linked Data: Evolving the Web into a Global Data Space (1st edition). Synthesis Lectures on the Semantic Web: Theory and Technology, 1:1, 1-136. Morgan & Clay-pool.